LLNL-CONF-665996

LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

Implementation of the Generalized Interaction Data Interface (GIDI) in the Mercury Monte Carlo Code

M. S. McKinley, B. R. Beck

January 8, 2015

M&C + SNA + MC 2015
Nashville, TN, United States
April 19, 2015 through April 23, 2015

ANS MC2015 - Joint International Conference on Mathematics and Computation (M&C), Supercomputing in Nuclear Applications (SNA) and the Monte Carlo (MC) Method • Nashville, TN • April 19-23, 2015, on CD-ROM, American Nuclear Society, LaGrange Park, IL (2015)

IMPLEMENTATION OF THE GENERALIZED INTERACTION DATA INTERFACE (GIDI) IN THE MERCURY MONTE CARLO CODE

M. S. McKinley and B. R. Beck
Lawrence Livermore National Laboratory
7000 East Ave., Livermore, CA 94551
mckinley9@llnl.gov; beck6@llnl.gov

ABSTRACT

Nuclear data formats that were defined 50 years ago are showing their age. The proposed replacement, Generalized Nuclear Data (GND), along with LLNL's interface library, Generalized Interaction Data Interface (GIDI), will replace the current nuclear data and collision package for LLNL's Mercury Monte Carlo radiation transport code. GND and GIDI are designed to support any project/target combination and more data types than the current formats and libraries. The initial implementation offers improved accuracy with respect to the evaluated data. Additional features need to be enabled before GIDI can fully replace the current collision package in Mercury.

Key Words: Monte Carlo, GIDI, GND, Mercury, nuclear data

1 INTRODUCTION

Around 1960, the Evaluated Nuclear Data Library (ENDL) [1] was developed at LLNL [2] as a format to store nuclear and atomic collision data for use in computer codes. Several years later, the Evaluated Nuclear Data Format (ENDF) [3] was developed to address the needs to store and exchange data across many organizations. While there have been some changes to these formats, they are showing their age. Issues like limited precision, human readability of the data, inline documentation, and inclusion of additional data fields have become problematic [4].

To address these issues, the Working Party on Evaluation Cooperation (WPEC) formed sub-group (SG38) with the title "Beyond the ENDF format: a modern nuclear database structure" [5]. The main task of SG38 is to design a new modern data format to replace ENDL and ENDF that will be used by all nuclear data institutions. The new format is currently called Generalized Nuclear Data (GND) [6]. GND stores data in a hierarchical structure that can be stored using XML or HDF5 [7]. The new format will support any projectile/target combination. Consistent nuclear masses, levels and lifetimes will be provided by an external particle database. Finally, the format will allow for units and interpolations to be stored with the data and will support multiple representations of the data (e.g., experimental, evaluated and processed representations) to be stored in a single file.

Mercury is a general-purpose Monte Carlo radiation transport code developed at LLNL. It currently tracks neutrons, gammas and five light charged particle isotopes (e.g., hydrogen, helium) [8]. Collision physics and nuclear data access are accomplished using the Monte Carlo All Particle Method (MCAPM) [9] library. MCAPM uses processed data derived from ENDL formatted data to compute cross sections and outgoing particle distributions. Since MCAPM is tied to ENDL data, it is also showing its age due to inflexibility that makes it hard to add new features.

Generalized Interaction Data Interface (GIDI) is a library that is being developed to access and sample the GND data, and serve it to a client code. Early versions of GIDI have been used in GEANT4 [10] since December of 2011 [11]. Recently, GIDI has been chosen to eventually replace MCAPM as the nuclear data access and collision simulation library for Mercury.

This paper will cover the progress of moving away from ENDL by looking at the history and how ENDL has been used. It will then cover the transition to GIDI and GND. The details of implementation will be handled next. Finally, some results from the Mercury using GIDI / GND will be presented.

2 FROM ENDL TO GND

2.1 Historical Progression of ENDL

The ENDL format was developed to represent binary collision data in a computer-readable format to facilitate neutronic calculations [1]. Despite its name, ENDL also handles atomic collisions and neutron-induced activation [12]. In its ASCII form, ENDL breaks down reaction data by incident particle, target isotope, reaction type, reaction property, and reaction modifier. Each breakdown, or data set, contains a short, concise header and a data block. The header consists of two lines that describe the target's half-life and mass, evaluation date, interpolation information, etc. There are no comments, and the data block may be up to 4 columns of fixed width. All data are pointwise.

The history of the ENDL data has not been well-preserved. While it started around 1960, little is known about where the data came from or how it progressed. Figure 1 shows the running sum of the number of changes or additions to the 1999 release of the ENDL (ENDL99) neutron library that was used at LLNL until 2008. As can be seen, ENDL99 consist of about 3,000 data sets, with more than ½ dating to before 1978.

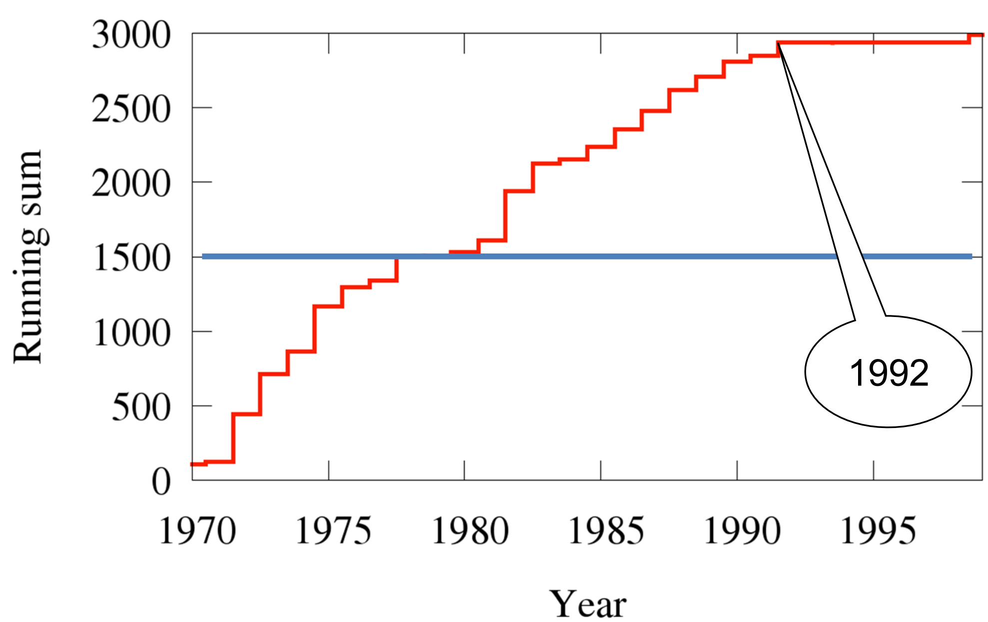

Figure 1. Running total of modifications and additions to ENDL99 neutron data.

From 1992 until 2008, only a few isotopes were modified. One of the major reasons for the slow updating of ENDL is the lack of sharing of data due to the differences in formats between ENDL and ENDF/B. Around 2000, an effort was started at LLNL to translate ENDF formatted data into the ENDL format. As part of this effort a coded named FETE (From ENDF To ENDL) [13] was developed. FETE would attempt to convert the data to LLNL's ENDL format, but there were issues along the way. Since ENDL only stores pointwise data, distribution parameters had to be expanded, which creates a file size versus data fidelity issue. ENDL also requires most distribution data to be in the lab frame; converting center-of-mass data to the lab frame significantly increase storage requirements. In addition, ENDL requires complete outgoing distributions for all trackable particles whereas ENDF/B does not.

Due to this effort, ENDL grew rapidly in number of isotopes and reactions as it incorporated data from ENDF/B-VII, JEFF, CENDL, JENDL, TENDL, ROSFOND, and IAEA Wolfram CRP. The ENDL neutron as projectile library grew from 110 target isotopes in ENDL99 to 585 and 918 isotopes in ENDL2009 and ENDL2011, respectively. This is in comparison to 423 isotopes found in ENDF/B-VII.1 neutron library [14].

While ENDL2011 has an impressive number of isotopes, it suffers from deficiencies of the ENDL format. There are limits on the precision of the data. The data are not easily human readable. Documentation tends to show up as extra files that are not well linked to the data. As new fields for the data are desired, there have been attempts to shoe-horn the new fields in. Particle masses from one evaluation to another may differ, which creates an inconsistent set of data. And reactions are pre-defined with enumerated reaction types called C numbers (which operate much like MT numbers in ENDF/B) that limit the ability to specify new and different reactions.

2.2 Processed Data

The basic ENDL data are not well suited to be used in transport codes. The data must be processed into a different form. This processing modifies the data (e.g. makes multigroup cross sections, heats nuclear data, compute transfer matrices) and stores it in a format designed for end use codes. The processed formats used at LLNL are listed in Table I [12].

Table I. Processed and auxiliary files and their uses

Processed File	Used for	Data representations
NDF	Deterministic transport	Multigroup data and transfer matrices
MCF	Monte Carlo transport	Multigroup and continuous energy data
TDF	Thermonuclear	Thermal averaged data (e.g., energy spectra, reaction rates)
Radchem	Radiation chemistry	Multigroup cross sections
BDFLS	Basic data lookup	Energy boundaries, weighting spectra, masses and half-lives

With all of these processed files, each with a different format, it is hard to maintain consistency in the output of a result. In Mercury, MCF, TDF and BDFLS may all be accessed in a single calculation. Most of the data tends to reside in the same directory, but since each may be specified separately, consistency is easily broken.

3 TRANSITIONING TO GND / GIDI

Since a new format has long been needed at LLNL to address the inadequacies of ENDL and the processed formats, around 2008 LLNL started developing a new format dubbed GND. In 2011, the new format became an international effort with the formation of OECD/NEA/WPEC/SG38 [5]. The GND format will also serve as a replacement for ENDF, which has about as long of a history as ENDL. By working with other agencies around the world for adoption, the acceptance of this format will lead to easier data sharing.

GND uses a hierarchical structure to represent the physics which may be stored using XML or HDF5. It will support any projectile and target. Consistent nuclear masses, excited energy levels, and lifetimes can be given by an external database. It will store units as well as interpolation type with the data. And the evaluated and processed data can be stored in the same format/location.

The FUDGE (For Updating Data and Generating Evaluation) infrastructure, which can translate ENDL and ENDF formatted data into GND, is freely available from [15]. Currently, FUDGE is used to reconstruct and heat cross section. It also is used to convert Legendre coefficients into pointwise data. In the future, GIDI will support heating of cross sections and convert Legendre coefficients into pointwise data. GIDI converts Madland/Nix parametric data into pointwise data when that data are read in. All other distribution types are sampled via their parametric data.

The role of the collision package in Mercury is to be fed in incident particle and target information and to return outgoing particles with their angles and energies. In the move to GND, Mercury will have to swap out its collision and nuclear data package, MCAPM, for the GND replacement, GIDI. In the intermittent time, both collision / data packages will be available at runtime. This will allow for ease of double checking results for verification and validation.

GIDI uses the Property of Particles (PoP) library that contains a particle database. The data for each entry includes mass, neutron count, proton count, nuclear level, and type of particle. This library is under development and can expand to hold more data for each particle. One of the advantages of this library is that it uses strings as names of the particles. This removes the constraint of using fake ZA numbers to identify different particles such as ZA of 1801 to represent hydrogen-1 in water for thermal neutron scattering data. The library can also return an internal index to avoid costly string comparisons for other access routines.

4 CURRENT STATE OF GIDI IN MERCURY

An initial implementation of GIDI has been installed into Mercury. It supports continuous energy and multigroup neutron transport. It also has delayed neutrons and photo-nuclear data implemented. However, photo-atomic data have not been added yet.

For testing, ENDF/B-VII.1 data were translated three different ways: 1) directly into GND for use via GIDI (GIDI/GND direct), 2) into ENDL via FETE. The ENDL data were translated into 2a) GND for use via GIDI (GIDI/GND via FETE) and 2b) into MCF processed file for use via MCAPM.

Two test cases were used to verify that GIDI was matching the results as obtained by MCAPM. Initially, the only isotope looked at was U-235, even though the rest of the isotopes are available.

The first test was a U-235 broomstick problem in which a semi-infinite thin wire of radius 10^{-5} cm is injected with mono-energetic 1 MeV neutrons that are all parallel and traveling down the wire. The density of the uranium was 2 g/cm^3 for this set of runs. This problem is used to test out first collision results since results of collisions leave the broomstick in the next event. Sometimes, there is a secondary collision on the way out. The density can be lowered or the radius shrunk to reduce this effect.

The number of collisions agreed very well among the three runs (GIDI/GND direct, GIDI/GND via FETE, and MCAPM). There was a minor difference in the elastic outgoing energy distribution. Since MCAPM uses equal probable bins to sample the angle, the results are represented by a histogram. GIDI currently samples directly from the probability density function and returns a smooth curve.

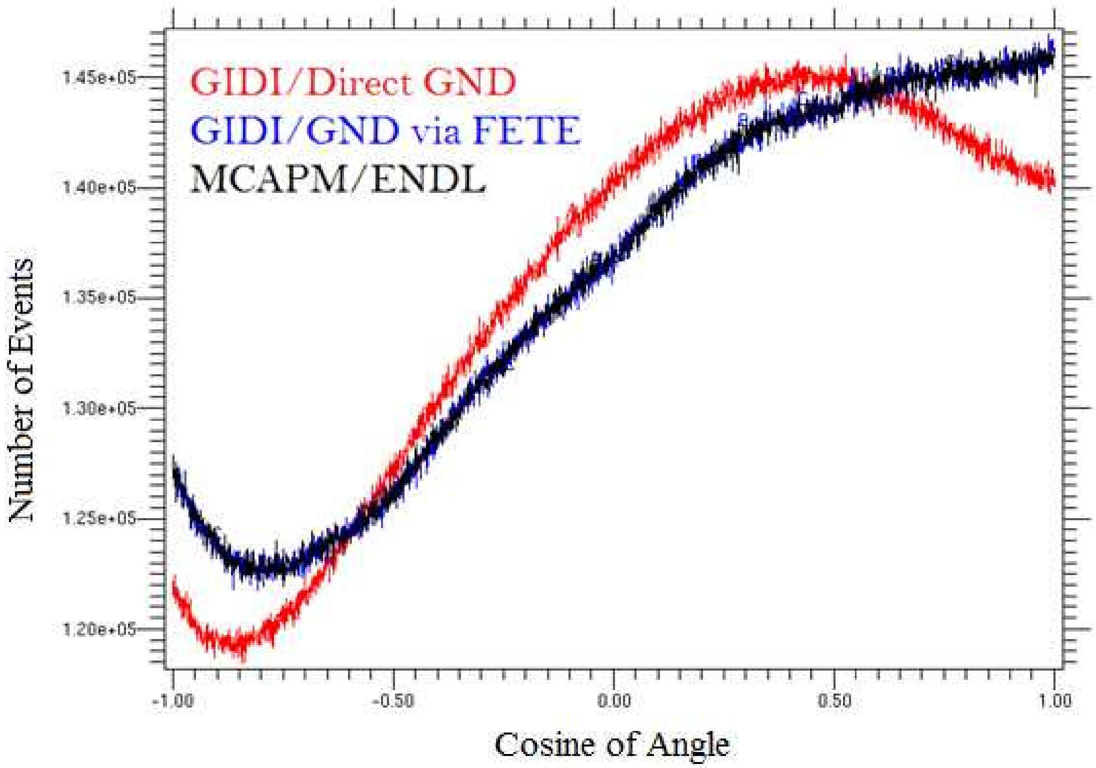

Figure 2. Comparison of the inelastic (n,n') reaction in a broomstick of U-235.

There was also a small difference in the shape of the inelastic angles sampled as seen in Figure 2. The angles are relative to the original flight path of the incident neutron, with cosine of the angle being 1.0 for a forward event and -1.0 for a backwards event. The outlier is the GIDI/GND direct. This difference is most likely due to FETE changing the ENDF Kalbach/Mann form of the data into ENDL pointwise data and not having enough angular points to accurately represent the form. If this is the case, then the "Direct to GND" curve is more correct sampling of the data. Further work will be required to verify this hypothesis.

A second test was to run a modified Godiva criticality benchmark that is a supercritical sphere of radius 8.7407 cm made entirely of pure U-235 at 18.74 g/cm^3 density. We looked at k_{eff} with delayed neutrons on and off. The results are shown in Table II. The standard deviation of the calculation was about 10^{-4}. There is good agreement between MCAPM and GIDI using both GND sets of data.

Table II. K_{eff} comparisons

Collision Package / Data Used	Prompt Criticality	Prompt + Delayed Criticality
MCAPM / ENDL via FETE	1.0214	1.0279
GIDI / GND via FETE	1.0211	1.0278
GIDI / GND Direct Translation	1.0211	1.0276

5 FUTURE WORK

There still remain many tasks left to fully implement GIDI into Mercury to have the same features that MCAPM offers. The first step is to verify that all isotopes are correct beyond U-235. The next step is to implement the special cross sections that Mercury uses for expected value tallies. These include the energy deposition, energy production, and gain “cross sections.” Following that, photo-atomic data needs implementation. This will involve making sure that a fixed grid cross section representation works with GIDI. And wrapping up the current trackable particles, we will make sure that light charged particles are implemented.

The previous tasks should hopefully be completed around June 2015. After that, point detector modeling will be implemented. This may take up to a year. Following that, we will allow for the user to modify cross sections at startup for their problem. And then there are several miscellaneous features that need to be implemented like optional relativistic effects, unresolved resonances, shared MPI memory, Gaussian distribution of the number of fission neutrons, and broadcasting the nuclear data with MPI. At this point, GIDI will have all of the needed features of MCAPM.

Along the way, GIDI has been targeted for next generation computers. It will be modified as needed to handle memory and heterogeneous systems. This is an active area of research which should not sacrifice serial runs for parallel efficiency.

6 ACKNOWLEDGMENTS

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

7 REFERENCES

1. Howerton, R.J., Dye, R.E., and Perkins, S.T., “Evaluated Nuclear Data Library,” Lawrence Livermore National Laboratory, Livermore, CA, UCRL-50400, **4**, Rev. 1, (1981).

2. Howerton, R.J., “A Review of Nuclear Reaction Data Evaluation in the U.S.”, Lawrence Livermore National Laboratory, Livermore, CA, UCRL-91926 (1985).
3. Trkov, A., Herman, M., and Brown, D.(eds.), "ENDF-6 Formats Manual: Data Formats and Procedures for the Evaluated Nuclear Data Files ENDF/B-VI and ENDF/B-VII", BNL-90365-2009 Rev.2, CSEWG Document ENDF-102 (2012).
4. Mattoon, C.M., and Beck, B.R., “WPEC SG38: Designing a New Format for Storing Nuclear Data,” ANS 2014 Annual Meeting, Reno, NV (not yet published) (2014).
5. “Beyond the ENDF format: A modern nuclear database structure,” http://www.oecd-nea.org/science/wpec/sg38/ (2014).
6. Mattoon, C.M., et. al., “Generalized Nuclear Data: A New Structure (with Supporting Infrastructure) for Handling Nuclear Data”, *Nuclear Data Sheets*, **113**, Iss.12, pp 3145-3171, ISSN 0090-3752 (2012).
7. “The HDF Group. Hierarchical Data Format version 5”, http://www.hdfgroup.org/HDF5 (2014).
8. Brantley, P.S., Dawson, S.A., et. al., “Mercury User Guide: Version d.16, LLNL-SM-560687 Mod #6,” Lawrence Livermore National Laboratory (2014).
9. Brantley, P.S., Hagmann, C.A., and Rathkopf, J.A., “MCAPM-C Generator and Collision Routine Documentation (Rev 1.2), Lawrence Livermore National Laboratory, Livermore, CA, UCRL-MA-141597 (2003).
10. Agnostielli, A., “GEANT4 – A Simulation Toolkit,” Nucl. Instr. And Meth. In Phy. Res. Sec. A: Acc, Spec, Det and Ass. Equip., **506**, Iss. 3, pp 250-303 (2003).
11. “Geant4 9.5 Release Notes,” http://www.geant4.org/geant4/support/ReleaseNotes4.9.5.html (2011).
12. Howerton, R.J., “A Status Report for Data Files and Monte Carlo Transport Codes Maintained by the Physical Data Group of LLNL,” Lawrence Livermore National Laboratory, Livermore, CA, UCID-19907 (1983).
13. Brown, D.A, et. al., "User's Guide to Fete: From ENDF To ENDL", Lawrence Livermore National Laboratory, Livermore, CA, UCRL-SM-218496 (2006).
14. Chadwick, M.B., et. al., “ENDF/B-VII.1 Nuclear Data for Science and Technology: Cross Sections, Covariances, Fission Product Yields and Decay Data,” *Nuclear Data Sheets*, **112**, Iss. 12, pp 2887-2996, LA-UR 11-05121 (2011)
15. “Generalized Nuclear Data (GND) format”, https://ndclx4.bnl.gov/gf/project/gnd/ (2014).